加微信：642945106 发送“ 赠送 ”领取赠送精品课程
发数字“ 2 ”获取众筹列表

04 | DevOps的衡量：你是否找到了DevOps的实施路线图?

2019-10-15 石雪峰

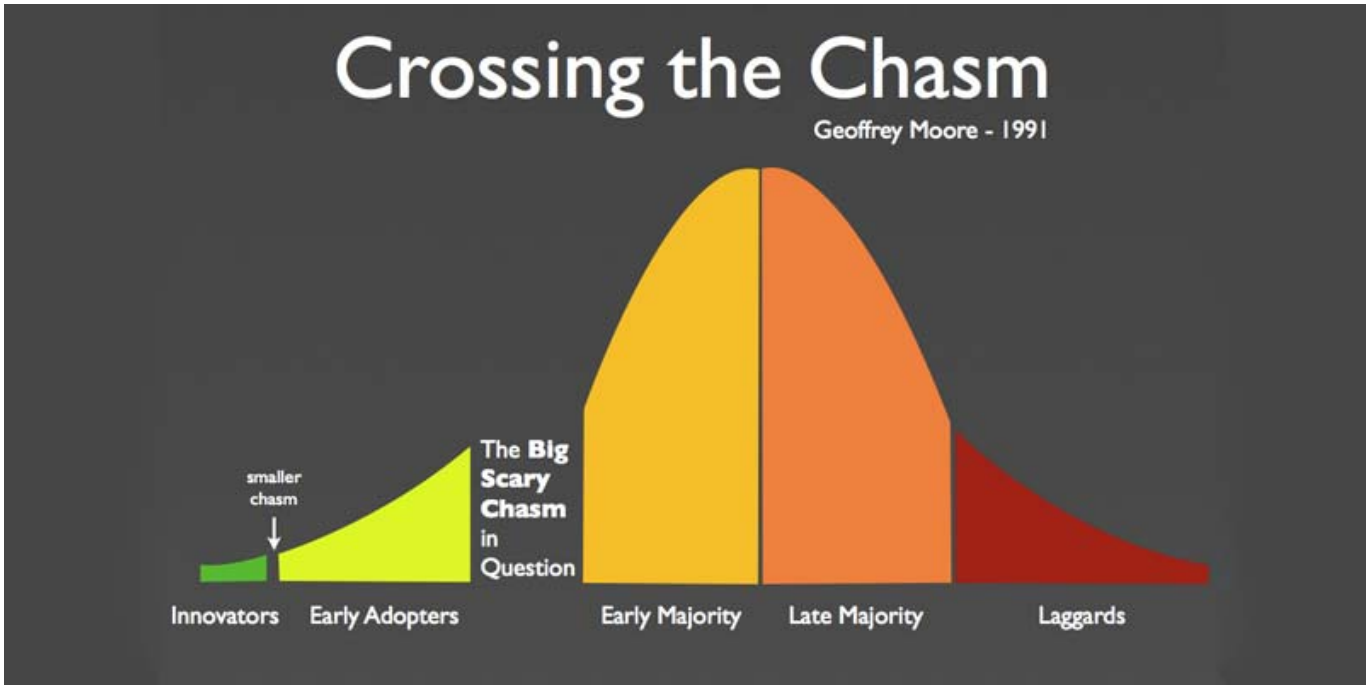
DevOps实战笔记

进入课程 >

讲述：石雪峰
时长 14:19　大小 13.12M

你好，我是石雪峰。今天我们来聊聊 DevOps 的实施路线图。

商业领域有一本特别经典的书，叫作《跨越鸿沟》，这本书中提出了一个“**技术采纳生命周期定律**”，对高科技行业来说，它的地位堪比摩尔定律。



简单来说，这个定律描述了一项新技术从诞生到普及要经历的 5 个阶段，这 5 个阶段分别对应一类特殊人群，即创新者、早期使用者、早期大众、晚期大众和落后者。这个定律表明，技术的发展不是线性的，需要经历一段蛰伏期，才能最终跨越鸿沟为大众所接受，成为业界主流。

当然，DevOps 这项所谓的新技术，在企业内部的落地也注定不是一帆风顺的。那么在这种情况下，你是否找到了 DevOps 的实施路线图呢?

从 2017 年第一届 DevOpsDays 大会中国站举办以来，DevOps 正式在国内驶入了发展的快车道。从一门鲜为人知的新技术思想，到现在在各个行业的蓬勃发展，各种思想和实践的激烈碰撞，DevOps 的理念和价值可谓是深入人心。

这样看来，DevOps 已经成功地跨越了技术发展的鸿沟，从早期使用者阶段进入了早期大众的阶段，而这也意味着越来越多的公司开始尝试 DevOps。

在 2017 年底，Forrester 的一组调查数据显示，将近 50% 的受访公司表示已经引入并正在实施 DevOps，30% 的公司表示有意向和计划来开启这项工作，而对 DevOps 完全不感兴趣的仅占 1%。可以说，2018 年就是企业落地 DevOps 的元年。

但是，就像你要去往一个未知的目的地时，需要导航帮你规划路径、实时定位，并在出现意外情况时及时提示你是否要重新规划路径一样，企业在实施 DevOps 的过程中，其实也面临着相似的问题。企业自身难以清晰定位 DevOps 的现状，客观评估 DevOps 相关的能力水平，识别当前所面临的最大瓶颈以及实施 DevOps 的阶段性成果预期……

回顾整个 IT 行业的发展历程，**新思想和新技术的发展，总是同标准化的模型和框架相伴相生的。**

我认为，**任何技术的成熟，都是以模型和框架的稳定为标志的。**因为当技术跨越初期的鸿沟，面对的是广大受众，如果没有一套模型和框架来帮助大众快速跟上节奏，找准方向，是很难大规模推广并健康发展的。

比如，软件开发领域的 CMMI 模型（软件能力成熟度模型）、运维行业的 ITIL 模型等，在各自的领域都久负盛名，甚至一度被各个领域的从业者奉为圭臬和行为准则，成为衡量能力高低的标尺。

我曾经在国内某大型通讯设备公司参与过 CMMI 评级项目。当时，就算业务压力再大，只要是关于通过评级的事情，所有部门都会高优先级支持。由此可见，整个公司都非常重视这个认证评级项目。

那么问题来了，在 DevOps 这项新思想和新技术不断走向成熟的过程中，是否也有类似的模型和框架，能够指导企业内部的 DevOps 转型落地工作呢？

防止断更 请务必加
首发微信：17161436655

答案是有的，而且有很多。只要你去谷歌上搜一下 DevOps 框架、模型等关键词，就能看到非常多的结果。尤其是国外的一些知名公司，比如 Atlassian、CloudBees、CA 等，基本上都有一套自己的模型和框架，来帮助企业识别当前的 DevOps 能力水平并加以改进。

我之前参与过工信部旗下的中国信息通讯研究院牵头制定的一套 DevOps 能力成熟度模型。这套模型覆盖了软件交付的方方面面，包括**敏捷开发管理**、**持续交付**和**技术运营**三大部分，同时，也有与应用架构设计、安全和组织结构对应的内容。

不仅如此，对于开发 DevOps 工具的企业来说，系统和工具模型更加偏向于平台能力，稍加整理就可以作为平台需求输入到开发团队中。目前已经有不少公司在参考这套模型进行 DevOps 实践。下图展示了这个模型的整体框架，如果你正在企业内部推进 DevOps 落地的话，可以参考一下。

能力类	一、研发运营一体化（DevOps）过程																
能力域	敏捷开发管理（标准2）			持续交付（标准3）							技术运营（标准4）						
能力子域	价值交付管理	敏捷过程管理	敏捷组织模式	配置管理	构建与持续集成	测试管理	部署与发布管理	环境管理	数据管理	度量与反馈	监控管理	事件与变更管理	配置管理	容量与成本管理	高可用管理	连续性管理	用户体验管理
能力项	需求工件	价值流	敏捷角色	版本控制	构建实践	测试分层策略	部署与发布模式	环境管理	测试数据管理	度量指标	监控采集	事件管理	运营配置管理	容量管理	应用高可用管理	应急管理	业务认知管理
	需求活动	仪式活动	团队结构	变更管理	持续集成	代码质量管理	持续部署流水线		数据变更管理	度量驱动改进	数据管理	变更管理		成本管理	数据高可用管理	风险管理	体验管理
						自动化测试					数据应用					危机管理	
能力类	二、研发运营一体化（DevOps）应用设计（标准5）																
能力类	三、研发运营一体化（DevOps）安全及风险管理（标准6）																
能力类	四、研发运营一体化（DevOps）组织结构（标准7）																
能力类	五、研发运营一体化（DevOps）系统和工具（标准8）																

步骤与原则

业界有这么多模型和框架，是不是随便找一个，直接照着做就行了呢？当然不是。

毕竟，每家企业所处的行业现状、竞争压力、市场竞争态势都不尽相同，组织架构、战略目标、研发能力、资源投入等方面也千差万别，很难有一条标准的路径，让大家齐步走。比如，同样是金融企业，让万人规模的大银行和百人规模的城商行同台竞技，本身就有点强人所难。

所以，在实际参考模型和框架的时候，我认为应该尽量遵循以下步骤和原则：

1.识别差距

从“道法术器”的角度来说，DevOps 的成熟度模型和框架处于“法”这个层面，也就是一整套实施 DevOps 的方法论，相当于是一幅战略地图，最重要的就是对 DevOps 实施所涉及到的领域和能力图谱建立全面的认知。

通过和模型、框架进行对标，可以快速识别出企业当前存在的短板和差距，并建立企业当前的能力状态基线，用于对比改进后所取得的效果。

2.锚定目标

数字化转型的核心在于**优化软件交付效率**。通过对标模型框架，企业需要明确什么是影响软件交付效率进一步提升的最大瓶颈，当前存在的最大痛点是什么，哪些能力的改善有助于企业达成预定的目标……同时，要根据企业的现状，甄别对标的差距结果，识别出哪些是真实有效的，哪些可以通过平台能力快速补齐。

比如，对于一家提供 CRM 软件的公司来说，容器化部署虽然在环境管理、部署发布等领域有非常多的优势，但并非当前的核心瓶颈和亟需解决的问题，那么就不应该纳入近期的改进

列表中。

通过现状分析，企业可以把有限的资源聚焦在那些高优先级的任务上，识别出改进目标和改进后要达到的预期效果。这些效果需要尽量**客观**和**可量化**，比如缩短 50% 的环境准备时长。

3.**关注能力**

模型和框架是能力和实践的集合，也就是道法术器的“术”这个层面，所以在应用模型的过程中，核心的关注点应该在能力本身，而不是单纯地比较数字和结果。

比如，亚马逊每天 23000 次部署的案例经常会被拿来举例子。这个数字的确相当惊人，但反过来想想，所有企业都需要达到这么高的部署频率吗？举个例子，一个客户端应用可以在几分钟内构建完成，但同样是构建，对于大型系统软件来说可能需要几个小时，那么到底多长时间才算达标呢？

我们不能只关注这些明星企业所达到的成就，而忽略了自身的需求。所以，**正确的做法是根据锚定的目标识别所需要的能力，再导入与能力相匹配的实践，不断强化实践，从而使能力本身得到提升**。

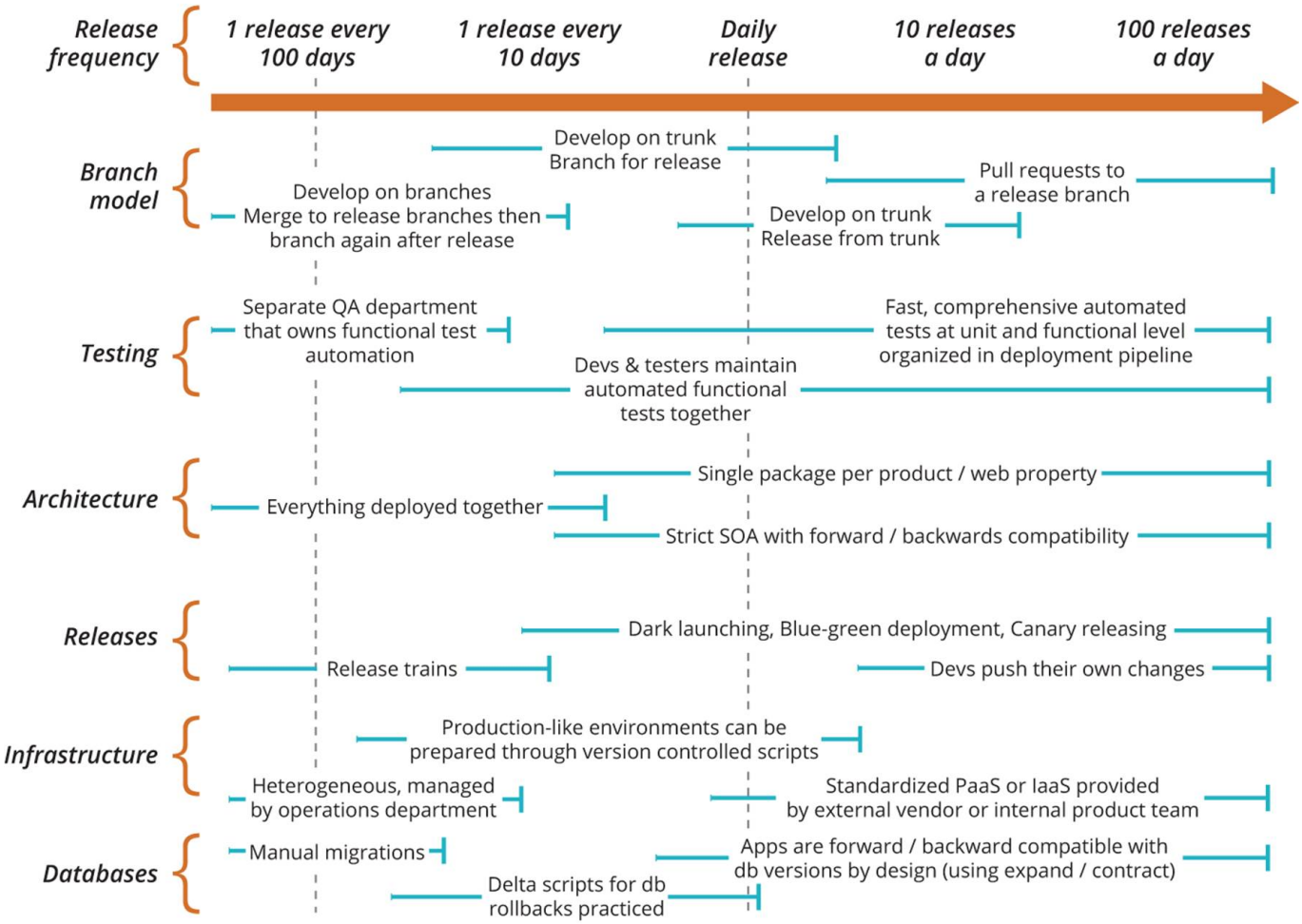
拼课微信：17161436665

4.**持续改进**

模型和框架本身也不是一成不变的，也需要像 DevOps 一样不断迭代更新，以适应更高的软件交付需要。另外，从今年的 DevOps 状态报告就可以看出，达到精英级别的比例从 2018 年的 7% 快速提升到 2019 年的 20%，也就是说，行业整体的能力也在不断提升，这就对企业的软件交付能力提出了更高的要求。

好了，以上这些就是我总结的企业应用 DevOps 能力模型和框架的步骤和原则。DevOps 作为一个系统性工程，同样需要与之配套的立体化实施方法，只有将方法、实践和工具结合起来，全方位推进，才有可能获得成效。

为了帮助你更好地理解 DevOps 实施的过程，我贴了一幅经典的部署引力图。



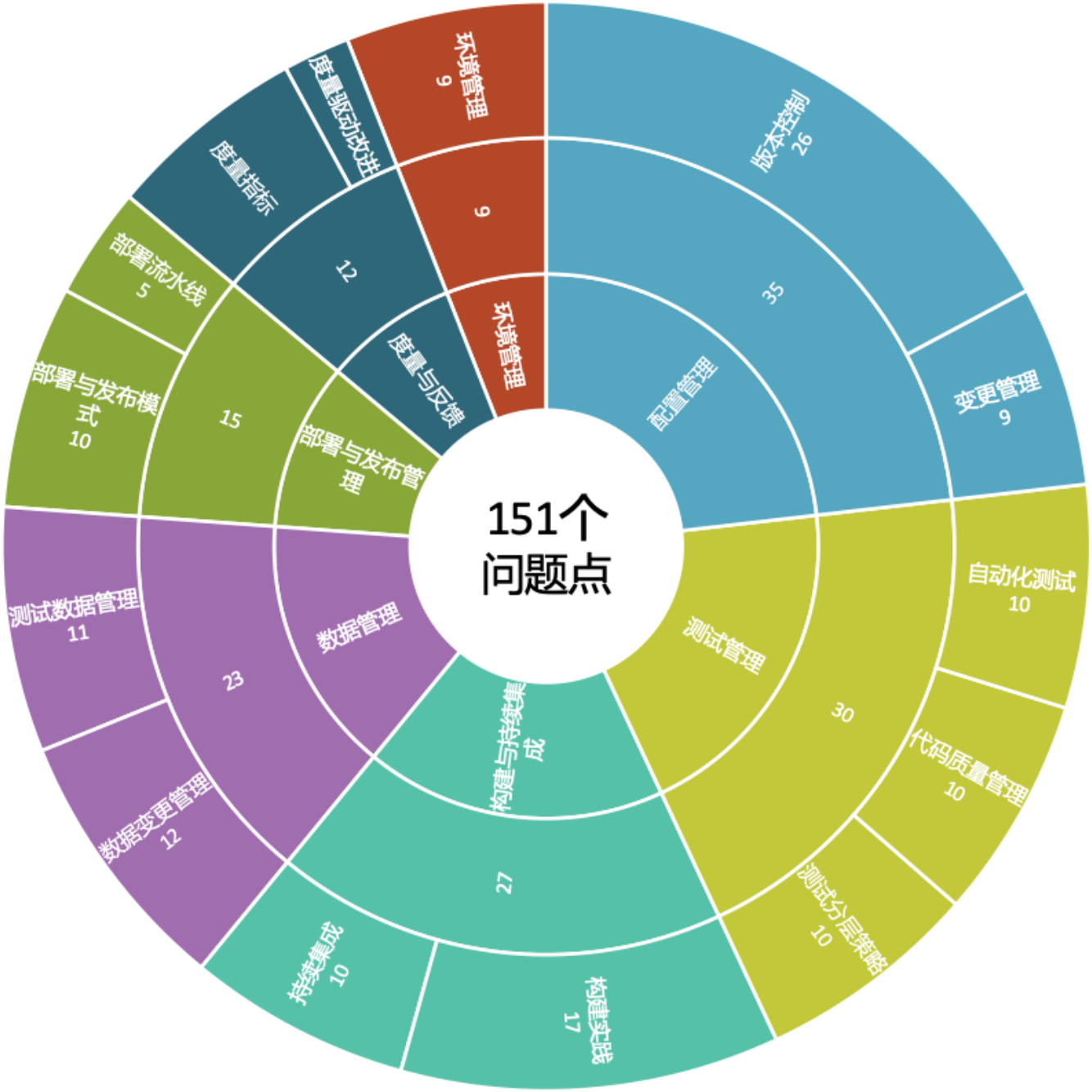
可以看出，当软件发布的频率从 100 天 1 次进化到 1 天 100 次的时候，分支策略、测试能力、软件架构、发布策略、基础设施能力，以及数据库能力都要进行相应的改动。比如分支策略要从长线分支变成基于特性的主干开发模式，而架构也要从大的单体应用，不断解耦和服务化。在实际应用中，企业涉及的领域甚至更多，因为这些仅仅是技术层面的问题，而组织文化方面也不可或缺。

实践案例

最后，我再跟你分享一个我之前参与改进的一个客户的案例。

刚开始跟这个客户交流的时候，他千头万绪，抓不准重点，甚至由于组织严格划分职责边界，基本上每讲到一块内容，他就要拉相应的人过来聊，在许多人都聊完之后，项目的全貌才被拼凑出来。我相信这并不是个例，很多公司其实都是如此。

于是，我们引入了能力成熟度模型，并基于模型对企业现有的能力水平进行了一次全盘梳理，并初步识别出了 100 多个问题点和 40 多个差距项。下面这张图就是汇总的大盘图，当然，部分数据进行了处理。



接下来，针对识别出来的这些差距点，我逐项跟企业进行了沟通，重点在于锚定一期的改进目标和具体工作事项。在沟通过程中，我发现由于企业所处行业的特殊性，或者客观条件不具备，有些内容并非优先改进事项，于是将改进事项缩减为 30 个，并识别出这些改进事项的相互依赖和预期目标。比如，这个企业之前初始化一套环境需要 2 周左右的时间，为了加快整体交付能力，我们将改进目标定到 1 周以内完成。

好啦，**有了改进目标和预期效果之后，就要分析哪些关键能力制约了交付效率的提升。**还拿刚才那个例子来说，核心问题在于**环境的初始化过程复杂**以及**审批流程冗长。**其中，原有的初始化过程是研发整理一份部署需求文档，来说明应用所依赖的环境和版本信息，并且这个需求还被整合到一个 40 多页的文档中。运维团队根据这个文档部署，每次都很不顺利，因

为软件功能迭代所依赖的环境也在不断更新，但文档写出来就再也没人维护了。所以，很多人说文档即过时，就是这个道理。

识别出核心能力在于**自动化环境管理**之后，团队决定引入基础设施即代码的实践来解决这个问题。关于具体的技术细节，我会在后面的内容中展开，这里你只需要知道，通过将写在文档中的环境配置说明，转变成配置化的信息，并维护在专门的版本控制系统中，从而使得基础环境的初始化可以在分钟级完成。

当然，审批环境的优化属于非技术问题，而是流程和组织方面的问题。当大家认识到这些审批在一定程度上制约了发布频率的提升，就主动改进了现有流程。针对不同的环境进行不同级别的审批，使得单次审批可以在当天完成。

这样优化下来，环境准备的时长大大缩短，从当初的 2 周缩短到了 2 天，改进效果非常明显。接下来，团队又识别出新的差距，锚定新的目标和预期效果，并且有针对性地补齐能力建设，走上了持续改进的阶段。

由此可以看出，**DevOps 的能力实践和能力框架模型相辅相成**：能力实践定义了企业落地 DevOps 的路线图和主要建设顺序，能力模型可以指导支撑方法的各类实践的落地建设；能力实践时刻跟随企业价值交付的导向，而能力模型的积累和沉淀，能够让企业游刃有余地面对未来的各种挑战。

至于 ITIL 和 CMMI，这些过往的框架体系自身也在跟随 DevOps 的大潮在持续演进，比如以**流程合规**为代表的 ITIL 最近推出了第 4 个版本。我们引用一下 ITIL V4 的指导原则，包括：**关注价值、关注现状、交互式流程和反馈、协作和可视化、自动化和持续优化、极简原则和关注实践。**

看起来是不是有点 DevOps 的味道呢？需要注意的是，DevOps 不会彻底颠覆 ITIL，只会在保证合规的前提下，尽可能地优化现有流程，将流动、反馈和持续学习改进的方法注入 ITIL 之中，从全局视角持续优化企业的价值交付流程。

总结

总结一下，今天我给你介绍了新技术和新思想的发展需要面对的鸿沟，而能力模型和框架是技术和思想走向成熟的标志，对于 DevOps 而言，也是如此。在面对诸多模型和框架的时候，企业需要立足自身，识别差异，锚定目标，关注能力，并持续改善软件的开发交付效

率。DevOps 的实施需要立体化的实施框架，通过模型、方法、能力和实践的相互作用，实现全方位的能力提升。

到此为止，我们整体介绍了 DevOps 的基本概念、核心价值、实施方法和路线图，帮助你建立了一套有关 DevOps 的宏观概念。接下来我们就会开始深入细节，尤其是针对每一项核心实践，我会介绍其背后的理念、实施步骤，以及所依赖的能力模型，手把手地帮助你真正落地 DevOps。

思考题

最后，给你留一道思考题：关于 CMMI、ITIL 和 DevOps，你觉得它们之间的关系是怎样的呢？企业该如何兼顾多套模型框架呢？

欢迎在留言区写下你的思考和答案，我们一起讨论，共同学习进步。如果你觉得这篇文章对你有所帮助，欢迎你把文章分享给你的朋友。

© 版权归极客邦科技所有，未经许可不得传播售卖。 页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇　03 | DevOps的实施：到底是工具先行还是文化先行？

下一篇　特别放送：成为DevOps工程师的必备技能（上）

精选留言 (12)

写留言

陈斯佳
2019-10-15

老师，我有个具体问题。我们公司现在的Jenkins Jobs都是使用在github上的Groovy/pipeline file生成的。每次新增一个新项目/应用，就多一个if else或switch case语句，弄的很复杂又不好维护。针对这种应用程序多、环境多的情况，您有没有什么好的方式来进行配置管理呢?

展开

作者回复: 所以你们也在实践Jenkins的pipeline是吧，两个具体的建议给到你。
第一，将job的配置代码化，也就是通过配置文件就能自动生成一个job，推荐你使用Openstack开源的Jenkins Job Builder工具，你在网上就能找到，使用起来非常方便。
第二，对于每一个项目获应用，使用独立的Jenkinsfile进行管理，相当于彼此独立，毕竟你把他们揉在一起没有意义。
当然你可能会问，那么大量重复的配置如何解决，其实这跟写代码模块化类似，我推荐你使用shares library来做一层抽象，把共同的东西提取出来进行复用哈。
关于Jenkins 的任何问题，欢迎你随时给我说哈！

3 9

刘超 mingo
2019-10-15

全都是理论，大哥，能讲点实践什么不？有么有资料让我提前预习下？

作者回复: 你好，目前就是基础理论篇呀😁 请问，你对具体哪方面的实践感兴趣呢，欢迎在留言区交流哈

3 3

Ops360
2019-10-18

在实施DevOps过程测试管理这个环节，如何有效的让测试和研发人员配合测试的一下工作，希望老师能分享单元测试和自动化测试方面的实施经验。在以前没有开展单元测试的研发团队，现在在原来的敏捷模式下给研发人员一项加新的工作内容，研发不愿意了，还有一些研发就没习惯单元测试，在测试人员实力不太强的团队，自动化测试应该从哪具体的小环节部分开始才合适?

展开

作者回复: 你好，你在留言中的两个字我觉得说的特别好，那就是习惯。我见过互联网公司单元测试完全推不下去的，也见过传统行业单元测试覆盖率能到达80%的，那么到底是人的能力问题，还是工具平台成熟度的问题呢？我觉得都不是，而是习惯的问题。
至于如何推行单元测试，除了开发团队自我要求特别高的情况以外，我见过的成功路径，都是自上而下行政命令式的推动执行的。但是，有两个小技巧，第一就是明确分工，就单元测试而言，有这么几部分工作，试点工具和框架，制定指标，写单元测试，自动化执行和提取数据，报告展示，这些都要有明确的分工职责，一般来说，资深开发人员和测试专家都可以负责工具和框架试点，大多数情况下业界的通用工具都是可以满足需求的。第二就是明确目标，循序渐进，不要试图一下子全面铺开，在小团队试点跑通跑成熟了，再推行，同时数据指标要明确，否则没有目标谁也不知道要做成什么样子。其实，单纯追求全面的覆盖率并不是好的建议，还是要根据业务场景，务实的推行，比如核心模块，基础组件比较适合，这些相信整个团队在认真单元测试这件事情的价值的前提下，都是可以识别出来的哈。

1

delly
2019-10-16

我觉得cmmi，itil和devops所解决的问题不一样，因为他们出现时整个软件行业的背景都不一样。比如运维团队再没有一个合适的框架之前，自己摸索，碰到问题只会自乱阵脚，有itil做指导，把一切都看作服务，服务交付的度量也有了，成功率也高了。而现在在成功率高的基础上，还要加快效率，所以devops这个理念就有了，慢慢交付不行，要持续交付。但老框架也会一直结合行业的变化对自身的内容做适当的增减，也慢慢的可以同时...
展开

作者回复: 你好，我非常赞同你的留言，其实公司里面一直在纠结一件事情，所谓业界最佳实践是否适用于企业自身，如果没有最佳实践漫无目的找不到方向，也不好统一大家认知，但是所有最佳实践也不能照单全收。所以最终还是需要相对有经验的人，带着最佳实践和公司业务团队的理解，去摸索出属于公司自己的最佳实践，这种循序渐进的方式才是DevOps应用的节奏吧。

1

leslie
2019-10-15

不知道这种理解是否有道理：其实现在的DevOps一定程度可以理解为软件研发过程中的ERP。

前天借着算法训练营开班的机会拜会了北京DB圈子里的同行；其实随着现在DB和OPS承担的服务器增长越来越恐怖，流程化、结构化反而成为现在软件开发过程中的问题。

软件的任何问题都是运维。。。DevOps其实一定程度就是就是软件内部开发和运维...

展开

作者回复: 这个比喻很有趣，实际上，运维的标准化程度要远远高于开发和测试，在加上运维工作本身就要求要紧的作风，所以在DevOps领域运维跑在前面也并不意外。那么下一步就是向前和向后拓展，向前拓展开发，向后拓展运营。

1

sugar

2019-10-16

看了这几章，终于有了点眉目，窥到了DevOps的一些斑点

作者回复: 你好，感谢你的留言，那么接下来，我们就要开始实践部分啦，也欢迎你提出你的问题，我们一起交流学习。

Geek_c991f2

2019-10-16

对于我们这些没有DEVOPS经验的工程师来说,理论还是抽象,想听下一个公司整个DEVOPS的实现流程,中间用到什么工具,哪些人员做了什么,先不用细化遇到什么问题,先有一个方向,老师,希望你能 看到我的恢复

作者回复: 你好，感谢你的回复，关于DevOps实施的整个流程，也是我会在第6讲中介绍到的。
其实，从我目前经历过的项目来看，大多数公司都是采用试点项目的方式来推进的，相当于一个改进的特区，从道法术器的角度来说，工具是最底层的，很多公司都有自己的工具链，单点能力也都足够强，所以最典型的DevOps平台就是持续交付流水线平台，通过这个平台串联软件交付的各个阶段，然后在具体的试点项目中进行落地，查漏补缺。
另外，除了打磨工具链之外，研发数据的度量也是试点的重要方面，通过识别出适合自己公司的度量指标，然后结合工具平台提取数据，展现趋势，是量化DevOps目标和结果的最客观的手段，通过试点项目，在团队中对指标达成共识，有了改进的基线，那么接下来要做的，就是引入实践，来证明改进的效果。
所以如果单纯来说用了哪些工具，我觉得比较容易泛泛而谈，就好比拿着锤子找钉子一样，而正确的姿势是，试点团队暴露出来的一些痛点，作为钉子，在打造好用的锤子，解决这个问题。
当然，除了以上我说的这些，对于试点团队来说，人的因素也非常重要，通过观察人的行为，制定相应的流程和制度，这些在未来大规模推广，都大有裨益。

Bryan Bai
2019-10-16

您这个DevOps成熟度模型挺有意思，有没有完整版参考？

作者回复: 你好，这个模型是业界公开的，任何人都可以看，你可以网上搜一下就能找到哈

陈文涛
2019-10-15

看来大家的留言，感觉作者应该开一堂课，京东在devops的实际落地，使用了哪些工具，各个岗位和环节是怎么衔接的。。。。。。：）

展开

作者回复: 哈哈，像京东这种公司的工具都是自己开发的，并不具备太多参考性哈，大公司都有这种习惯，什么都要自己做才行😄

3

Robert小七
2019-10-15

目前的工作是给一些金融行业的客户做devops咨询，帮助企业改进研发效能！希望和老师一起共同探讨devops的落地实施！

展开

作者回复: 你好，看来是半个同行，不知道你在指导企业转型的过程中是否也有一套模型框架呢。是否也有一些新法可以拿来讨论的呢？期待你的更多回复哈

1

风过留痕
2019-10-15

希望老师在今后的分享中能够更加细致的讲解原理和结合实际情况的实践，期待您的下一讲。

作者回复: 你好，感谢你的回复，收到你的反馈。我也提个小小的期望，短短的10几分钟的课程难以面面俱到，希望你能提出你遇到的实际问题和思考，我们可以在留言区跟大家一起交流讨论，营造一种共同学习的氛围哈！

天天向上

2019-10-15

问题点的详细清单能给下吗?

展开

作者回复: 你好，你是指案例中的问题列表吗，其实这些都是针对具体公司给出来的，可能没有普遍意义。比如，将构建脚本纳入版本控制系统，增加平均故障修复时间的统计，这些都是在对标之后给出来的改进建议哈。